Robust Infrared Vehicle Tracking across Target Pose Change using L_1 Regularization

Haibin Ling[1], Li Bai[2], Erik Blasch[3], and Xue Mei[4]
[1]Computer and Information Science Department, Temple University, Philadelphia, PA U.S.A.
[2]Electrical and Computer Engineering Department, Temple University, Philadelphia, PA U.S.A.
[3]Air Force Research Lab/SNAA, OH, U.S.A.
[4]Intel Corporation, U.S.A.
hbling@temple.edu, lbai@temple.edu, erik.blasch@afosr.af.mil, nathanmei@gmail.com

Abstract - In this paper, we propose a robust vehicle tracker for Infrared (IR) videos motivated by the recent advance in compressive sensing (CS). The new eL$_1$-PF tracker solves a sparse model representation of moving targets via L_1 regularized least squares. The sparse-model solution addresses real-world environmental challenges such as image noises and partial occlusions. To further improve tracking performance for frame-to-frame sequences involving large target pose changes, two extensions to the original L_1 tracker are introduced (eL$_1$). First, in the particle filter (PF) framework, pose information is explicitly modelled into the state space which significantly improves the effectiveness of particle sampling and propagation. Second, a probabilistic template update scheme is designed, which helps alleviating drift caused by a target pose change. The proposed tracker, named eL$_1$-PF tracker, is tested on IR sequences from the DARPA Video Verification of Identity (VIVID) dataset. Promising results from the eL$_1$-PF tracker are observed in these experiments in comparison with previous mean-shift and original L_1–regularization trackers.

Keywords: Visual tracking, particle filter, L_1-regularization, Infrared target tracking

1 Introduction

Visual tracking is a critical task in many military-specific and security/medical related computer vision applications such as surveillance, robotics, human computer interaction, vehicle tracking, and medical imaging, etc. The challenges in designing a robust visual tracking algorithm include: occlusions, presence of noise, varying viewpoints, background clutter, and illumination changes [25]. For years, researchers have proposed a variety of tracking algorithms to overcome these difficulties. Most existing tracking methods generally consist of two components: an *inference framework* (e.g., Kalman filter, particle filter, etc.) and a *target representation* (e.g., linear subspace, sparse representation, etc.). A summary of related work is given in Section 2 and a thorough survey for general visual tracking can be found in [25].

In this paper we focus on vehicle tracking in infrared videos. Compared with normal videos, infrared videos are robust to day/night changes and hence suitable for environments with poor or unstable lighting conditions. Despite the advantage, infrared data brings to visual tracking additional challenges. One such challenge lies in the low image quality, which usually involves blurring and noises. Another difficulty is the background-foreground similarity, which is usually due to the relatively low contrast in pixel intensity. In our work where vehicle traffic videos are taken from air, pose variation becomes an important challenge. Note that the change of vehicle poses also brings serious changes in appearance. Some example frames from the DARPA Video Verification of Identity (VIVID) dataset [21] are shown in Figure 1, where the challenges of low image quality and large-pose change can be clearly observed.

Figure 1. Example frames from IR videos in the VIVID dataset [21]. Top row: challenges in background-foreground similarity and low image quality. Bottom row: pose changes.

In our recent work [17], a robust visual tracker, named *L_1–PF* tracker, is introduced that takes advantage of recent advances in compressive sensing and sparse representation [2] [6]. During tracking, a target candidate is represented by a linear combination of template sets composed of both target templates, which are obtained from previous frames, and trivial templates, each of which contains only one nonzero pixel. The intuition is that a good candidate target should be efficiently represented by

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
JUL 2010		**00-00-2010 to 00-00-2010**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Robust Infrared Vehicle Tracking across Target Pose Change using L1 Regularization	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Temple University,Computer and Information Science Department,Philadelphia,PA,19122**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

Presented at the 13th International Conference on Information Fusion held in Edinburgh, UK on 26-29 July 2010. Sponsored in part by Office of Naval Research, Office of Naval Research Global, and U.S. Army Research Laboratory's Army Research Office (ARO).

14. ABSTRACT

In this paper, we propose a robust vehicle tracker for Infrared (IR) videos motivated by the recent advance in compressive sensing (CS). The new eL1-PF tracker solves a sparse model representation of moving targets via L1 regularized least squares. The sparsemodel solution addresses real-world environmental challenges such as image noises and partial occlusions. To further improve tracking performance for frame-toframe sequences involving large target pose changes two extensions to the original L1 tracker are introduced (eL1). First, in the particle filter (PF) framework, pose information is explicitly modelled into the state space which significantly improves the effectiveness of particle sampling and propagation. Second, a probabilistic template update scheme is designed, which helps alleviating drift caused by a target pose change. The proposed tracker, named eL1-PF tracker, is tested on IR sequences from the DARPA Video Verification of Identity (VIVID) dataset. Promising results from the eL1-PF tracker are observed in these experiments in comparison with previous mean-shift and original L1? regularization trackers.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**8**	

the target templates and therefore requires a sparse combination coefficient vector. Such a representation is then effectively solved by L_1 regularization [2] [6]. Once a target is determined, the target tracking is continued using a particle filter (PF) framework. While the L_1-PF tracking method has been shown to be robust in general visual-spectrum tracking scenarios, its application to IR videos with rotation change has not been exploited.

In this paper, we extend the L_1-PF tracking method [17] with two major extensions for vehicle tracking in IR videos. First, we use five state parameters (two for dimension, one for rotation, and two for position) to explicitly describe the tracking state, while in [17] the six dimensional affine states are used. In comparison, the *extended L_1-PF* (eL_1-PF) method provides more flexible control over large frame-to-frame changes in target pose. Meanwhile, the sampling in the new state space becomes more reliable since it's one dimensional lower than that in [17]. Second, we use probabilistic template update scheme, while in [17] the updating is biased toward template set sparseness. We show that the new update scheme is more reliable to target rotation and pose changes. The proposed method is tested on several IR videos from the VIVID dataset [21]. In all these videos, our method demonstrated excellent performance in comparison with other state-of-the-art trackers (i.e. mean-shift and original L_1–PF trackers).

The rest of the paper is organized as follows. In Section 2 we summarize related research on visual tracking. Then, in Section 3, we present the particle filter framework with our *pose-based* state representation. After the sparse appearance representation is introduced, we present our new *template-update* scheme in Section 4. Section 5 describes the experiments. Finally, conclusions are made in Section 6.

2 Related Work

Target tracking is an important topic in computer vision and it has been studied for several decades. In this section, we summarize studies that are related to our work; however, a thorough survey is given in [25]. For visual tracking, robust similarity measures have been applied and the *mean-shift* (MS) algorithm or other optimization techniques utilized to find the optimal solution [5]. The MS algorithm iteratively carries out a kernel- based search starting at the previous location of the object. The success of the mean shift depends highly on the discriminating power of the histograms that are considered as the objects' probability density function.

Tracking can be considered as an estimation of the state for a time series state space model. The problem is formulated in probabilistic terms as an *inference framework*. Works use a Kalman filter [2][23][23] to provide solutions that are optimal for a linear Gaussian model. The *particle filter* (PF), also known as the sequential Monte Carlo method [7] is one of the most popular approaches. The PF recursively constructs the posterior probability density function of the state space using Monte Carlo integration. PF has been developed in the computer vision and information fusion community and applied to tracking problems also under the name Condensation [11]. In [26], an appearance-adaptive model is incorporated in a particle filter to realize robust visual tracking and classification algorithms. In [12], an efficient method for using subspace representation in a particle filter by applying Rao-Blackwellization to integrate out the subspace coefficients in the state vector.

The *target representation* used in our eL_1-PF method is related to subspace-based tracking methods. These methods treat tracking as searching for a target in a subspace learned from previous tracking results. For example, in [1] the appearance of the object is represented using an eigenspace. The object appearances are represented using affine warps of learned linear subspaces of the image space [10]. In [18], a tracking method is proposed to incrementally learn a low-dimensional subspace representation, efficiently adapting online to changes in the target appearance.

Our study is largely inspired by recent advances in compressive sensing [6] and its applications to many pattern recognition tasks (e.g., face recognition [22], shadow modeling [16], etc.). The problem is to exploit the compressibility and sparsity of the true signal and use a lower sampling frequency than the Shannon-Nyquist rate. Sparsity leads to efficient estimation, efficient compression, dimensionality reduction, and efficient modeling. In particular, the proposed method is a direct extension of our previous work [17] on L_1 tracking. Details of L_1 tracking will be illustrated in the following sections.

3 Pose Tracking in the Particle Filter Framework

In this section we first give a brief overview of the particle filter (PF) framework. After that, we will describe the pose-based state space and observation model used in our eL_1-PF tracking algorithm.

3.1 Particle Filter

We use the particle filter (PF) framework to guide the tracking process. PF [7] is a well known Bayesian sequential importance sampling technique used for posterior distribution estimation of state variables in a dynamic system. PF becomes a popular tracking framework since the seminal work in [11] for nonlinear, non-Gaussian environments. The framework contains mainly two iterative steps: (1) a *prediction step* that is used to predict the target in the current frame based on previous observations, and (2) an *update step* that maintains sample (particle) weights for the Bayesian inference.

In the context of visual tracking, let $\{y_1, y_2, \ldots\}$ be the observations (e.g., appearance in video frames) and $\{x_1,$

x_2, …} be the states (e.g., poses of target objects in the video), where y_t and x_t denote the observation and state variable at time t respectively. The task of prediction is to estimate the distribution of x_t given all previous observations $y_{1:t-1}=\{y_1, y_2, \ldots, y_{t-1}\}$ up to time t-1. This conditional distribution can be recursively computed as

$$P(x_t \mid y_{1:t-1}) = \int P(x_t \mid x_{t-1}) P(x_{t-1} \mid y_{1:t-1}) dx_{t-1} . \quad (1)$$

The update step, at time t, updates the posterior probability $P(x_t|y_{1:t})$ through Bayes' rule,

$$P(x_t \mid y_{1:t}) = \frac{P(y_t \mid x_t) P(x_t \mid y_{1:t-1})}{P(y_t \mid y_{1:t-1})}, \quad (2)$$

where $P(y_t|x_t)$ indicates the observation likelihood.

The posterior $P(x_t|y_{1:t})$ is in general very difficult to be computed explicitly due to the integration in formula (1). In the particle filter framework, it is instead approximated by a sample set $\{x^t_1, x^t_2, \ldots, x^t_N\}$ with importance weights $\{w^t_1, w^t_2, \ldots, w^t_N\}$. Note that both samples and weights are time dependent. The candidate samples x^t_i are drawn from an importance distribution and the weights are updated accordingly. Then the samples are resampled to generate a set of equally weighted particles according to their importance weights and to avoid degeneracy.

The focus of this paper is not in PF algorithms but in sparse representation. For this reason, we follow the standard CONDENSATION framework in [11].

3.2 Modeling Object Pose in Particles

In [17] affine particles are used to capture the change of target shape due to pose and view-point changes. Specifically, a state x contains six elements from the affine matrix. A drawback of such a formulation lies in the difficulties to control track accuracy over poses, especially about rotation, since rotation is not explicitly expressed in the six dimensional affine transformation vectors.

To alleviate this problem, in this work we treat a target region with a rectangle and explicitly model its pose using five parameters. In particular, a state $x=(w,h,\theta,p_1,p_2)'$ is composed of the target width, height, rotation angle, and center-of-gravity position (x,y) respectively. With the formulation, the state transition probability is explicitly modeled with a Gaussian distribution,

$$\begin{aligned} P(x_t \mid x_{t-1}) &= \Phi(x_t - x_{t-1}; \Sigma) \\ &= \frac{1}{(2\pi)^{5/2} |\Sigma|^{1/2}} \exp\{-\frac{1}{2}(x_t - x_{t-1})' \Sigma^{-1} (x_t - x_{t-1})\} \end{aligned} \quad (3)$$

where Φ is the zero mean Gaussian distribution with covariance matrix $\Sigma=\mathrm{diag}(\sigma^2_w, \sigma^2_h, \sigma^2_\theta, \sigma_1^2, \sigma^2_2)$ such that $\sigma_w=\sigma_h$ are used for target dimension, σ_θ for rotation and $\sigma_1=\sigma_2$ for position. It is worth noting that the pose parameterization can also be helpful to model object motion.

Another advantage of the new model is that, under the same condition, it requires fewer particle samples than does the affine model. This is due to the reduced state dimension. Though the reduction is only one dimension, the number of samples saved can be significant since they are usually exponential in state dimensions.

3.3 Observation Model

With a state vector x_t, a rectangle region can be cropped from the frame at time t. This region is then normalized to have the same size as the templates. In addition, the intensities inside the region are normalized to have zero-mean and unit variance, which is known to be robust to affine lighting change. After that, the pixel intensities in this rectangle are concatenated to form the observation vector y_t.

We model the observation likelihood $P(y_t|x_t)$ so that it captures the similarity between a target candidate and the target templates. As will be described in next section, we use the subspace spanned by the target template set to model the observation appearance. The observation likelihood is then modeled as a Gaussian distribution over the approximation residual

$$\begin{aligned} P(y_t \mid x_t) &= \Phi(r(y_t); \sigma_r) \\ &= \frac{1}{(2\pi)^{1/2} \sigma_r} \exp\{-\frac{r(y_t)^2}{2\sigma_r^2}\} \end{aligned}, \quad (4)$$

where $r(y_t)$ is residual defined in Section 4 and σ_r denotes the variance.

4 Tracking via L_1 Regularization

4.1 Sparse Representation for Visual Tracking

In our recent work [17], a robust visual tracker is proposed using sparse representation and L_1 regularization. In the method, a moving target is approximated by the linear subspace spanned by a template set. Instead of using a transferred low-dimensional space, as in many previous studies, we treat the target in the new frame as a sparse representation using previous observed targets. Intuitively, a new target should not deviate too much from its previous observations, and it should require only a limited number of previous instances to model its appearance.

4.1.1 Template-based Target Representation

Let a candidate target be $\mathbf{y} \in \Re^d$ (we stack columns of a candidate patch to form a 1D vector), and let the template set be $\mathbf{T}=[\mathbf{t}_1 \ldots \mathbf{t}_n] \in \Re^{d \times n}$ ($d >> n$) contain n target templates

achieved from previous frames. Note that bold symbols are used to emphasize vectors or matrices. The global appearance of one object under different illumination and viewpoint conditions is known to lie approximately in a low-dimensional subspace. Therefore, we have,

$$\mathbf{y} = \mathbf{T}\mathbf{a} + \quad = a_1\mathbf{t}_1 + a_2\mathbf{t}_2 + \ldots + a_n\mathbf{t}_n + \quad , \qquad (5)$$

where $\mathbf{a}=(a_1, a_1, \ldots, a_n)' \in \Re^n$ is called a *target coefficient vector*, and denotes the approximation error.

In many visual tracking scenarios, target objects are often corrupted by noise or partially occluded in an image. Occlusions create unpredictable errors, affecting any part of the image and appearing at any size on the image. To incorporate the effect of occlusion and noise, similar to a previous study in face recognition [22], Equation 1 can be rewritten as

$$\mathbf{y} = a_1\mathbf{t}_1 + \ldots + a_n\mathbf{t}_n + e_1\mathbf{i}_1 + \ldots + e_n\mathbf{i}_n + e_{-1}\mathbf{i}_{-1} + \ldots + e_{-n}\mathbf{i}_{-n}$$

$$= [\ \mathbf{T}\ \ \mathbf{I}\ ] \begin{bmatrix} a \\ e \end{bmatrix} = \mathbf{B}\,\mathbf{c}\ , \qquad (6)$$

where $\mathbf{i}_k \in \Re^d$ is a vector with only one nonzero entry $\mathbf{i}_k(j)=_{kj}$, and $\mathbf{i}_{-k}=-\mathbf{i}_k$ i.e. $\mathbf{I}=[\mathbf{i}_1\ \mathbf{i}_2\ \ldots\ \mathbf{i}_n\ \mathbf{i}_{-1}\ \mathbf{i}_{-2}\ \ldots\ \mathbf{i}_{-n}]$ is a concatenation of an identity matrix and a negative identity matrix. We call $\mathbf{i}_k$ *trivial templates* and $\mathbf{e}=(e_1\ e_2 \ldots e_n\ e_{-1}\ e_{-2} \ldots e_{-n})' \in \Re^{2n}$ *trivial coefficients*. Figure 2 illustrates the representation, where a target candidate (the region in the red rectangle) is represented by a linear combination of templates.

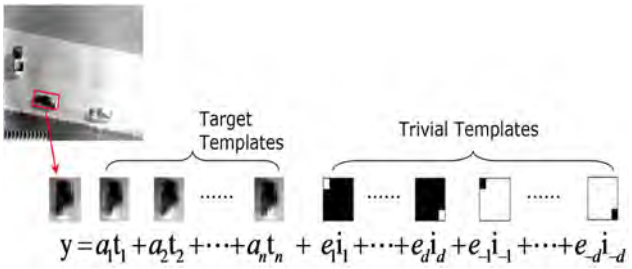


Figure 2. Representing a candidate target **y** with target template set **T** and trivial templates **I**.

4.1.2 Sparse Approximation and L_1 Regularization

Intuitively, a good target candidate is efficiently represented by the target templates, which implies a sparse coefficient vector **c**. In other words, most trivial coefficients tend to vanish. In the case of occlusion (and/or other unpleasant issues such as noise corruption or background clutter), a limited number of trivial coefficients will be activated, but the whole coefficient vector remains sparse. A poor target candidate, on the contrary, often leads to a dense representation. An example of such phenomena is shown in Figure 2. The sparse representation is achieved through solving an L_1-regularized least squares problem, which can be done efficiently through convex optimization. Then the candidate with the smallest target template projection error is chosen as the tracking result. After that, tracking is led by the Bayesian state inference framework in which a particle filter is used for propagating sample distributions over time.

The system model in (6) is underdetermined and does not have a unique solution for **c**. The error caused by occlusion and noise typically corrupts a fraction of the image pixels. Therefore, for a good target candidate, there are only a limited number of nonzero coefficients in **e** that account for the noise and partial occlusion. Consequently, we want to have a sparse solution that requires an L_0 regularization:

$$\min \|\mathbf{B}\mathbf{c} - \mathbf{y}\|_2^2 + \lambda \|\mathbf{c}\|_0 . \qquad (7)$$

However, L_0 regularization is in general a tough problem. Fortunately, thanks to the recent advances in sparsity analysis [6][2], it can be well approximated via an L_1 regularization. With this observation, our task becomes an L_1-regularized least squares problem:

$$\min \|\mathbf{B}\mathbf{c} - \mathbf{y}\|_2^2 + \lambda \|\mathbf{c}\|_1, \qquad (8)$$

$$\text{subject to} \quad \mathbf{c} \geq 0,$$

where $\|.\|_1$ and $\|.\|_2$ denote the L_1 and L_2 norms respectively, λ is the regularization parameter, and $\mathbf{c} \geq 0$ is defined element-wise.

To solve the L_1-regularized least squares problem (8), we use the algorithm in [13] that is an interior-point method. The method uses the preconditioned conjugate gradients (PCG) algorithm to compute the search direction and the run time is determined by the product of the total number of PCG steps required over all iterations and the cost of a PCG step.

We then find the tracking result by finding the smallest residual after projecting on the target template subspace. Specifically, at frame *t*, let $\boldsymbol{X}=\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$ be the *m* state candidates and $\boldsymbol{Y}=\{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_m\}$ be the corresponding observations, the tracking result $\mathbf{y}'$ is chosen by

$$\mathbf{y}' = \text{argmax}_{\mathbf{y} \in \boldsymbol{Y}}\, r(\mathbf{y}) , \qquad (9)$$

where $r(\mathbf{y})=\|\mathbf{y}-\mathbf{T}\mathbf{a}\|$ is the approximation residual with respect to the current template set **T** and template coefficients **a** achieved by solving (8). Note, in the above and all following notations, the frame id *t* is dropped for clarity, whenever there is no ambiguity.

4.1.3 Advantages of Using L_1 Regularization

The L_1 tracker presents many potential benefits:

1. *Robust to occlusion*. Unlike component analysis, the sparsity constraints provide an automatic means to "filter" out local occlusions. Such method has been

successfully applied to handle partial occlusions in face recognition [22].

2. *Robust to appearance change* due to view change, illumination variation, or noise.
3. *Easy to generalize*. The proposed representation does not require object- or class-specific modeling. Consequently, it provides a general solution that applies readily to different objects. For example, in [17], the tracker is has been applied to track different targets including faces, human bodies, vehicle, etc.

4.2 Probabilistic Template Update

Template-based methods have been studied for several decades in the computer vision literature, dating back at least to 1981 [14]. The object is tracked through the video by extracting a template from the first frame and finding the object of interest in successive frames. It has been observed that fixed template sets are not sufficient to handle variations temporal data [15], which is particularly true for our task where targets undergo large pose changes.

Intuitively, object appearance remains the same only for a certain period of time, but eventually the template is no longer an accurate model of the object appearance. If we do not update the template, the template cannot capture the appearance variations due to illumination or pose changes. If we update the template too often, small errors are introduced each time the template is updated. The errors are accumulated and the tracker drifts from the target. We tackle this problem by dynamically updating the target template set **T**.

In [17], a *template-update scheme* (TUS) is proposed in favour of maintaining a diverse template set. Template updating is triggered when a new target is found to be rather different than all existing templates. In experiments on Infrared (IR) videos when targets have large pose changes, however, we found such a TU scheme vulnerable to drift problems. The main reason is that, in IR videos, the intensity patterns of the target of interest are often similar to those in the surrounding regions (see Figure 1 for examples). The similar target-to-background intensity makes a slow updating (due to target extraction and numerous templates) vulnerable to drift problems. On the other hand, we argue that diversity of template sets does not contribute much to the approximation accuracy, since the L_1 regularization already takes care of redundancy.

Based on the above observation, we propose instead a different template update scheme based on the *observation likelihood*. Intuitively, the new scheme updates the template set only when the system feels confident enough in the new target. Specifically, for a newly detected target **y**, the update happens when its observation likelihood $P(\mathbf{y}|\mathbf{x})$ is greater than a threshold .

In the scheme, template set **T** is associated with a weight vector $\mathbf{w}=(w_1, w_2, \ldots, w_n)$. These weights are used (1) to control the importance of each template $\mathbf{t}_i$ (by forcing $\|\mathbf{t}_i\|=w_i$) and (2) to choose the least useful template for replacing. The template weights are also dynamically updated, but we use a different update scheme than in [17]. In particular, the update is governed by the observation probability $P(\mathbf{y}'|\mathbf{t}_i)$ given a template

$$P(\mathbf{y}'|\mathbf{t}_i) = \Phi(\| \mathbf{y}' - \mathbf{t}_i \|; \sigma_r) \qquad (10)$$

$$= \frac{1}{(2\pi)^{1/2}\sigma_r} \exp\{-\frac{\| \mathbf{y}' - \mathbf{t}_i \|^2}{2\sigma_r^2}\},$$

where $_r$ is the variance as in the observation likelihood. The new template update algorithm is summarized in Table 1.

Table 1. Probabilistic Template Update

Input:

y' is the newly chosen tracking target.
x is the chosen state corresponding to **y**'.
a is the solution to (8).
w is current weights, such that $\|\mathbf{t}_i\|=w_i$.
τ is a predefined threshold.

Output:

Updated template set **T** and associated weights **w**.

```
1.  for i=1..n
2.    w_i ← w_i + P(y'|t_i)     /* update weights */
3.  endfor
4.  if (P(y'|x) > τ)
5.    j ← argmin_i w_i
6.    t_j ← y'                  /* replace an old template */
7.    w_j ← median(w)           /* replace an old weight */
8.  endif
9.  Normalize w such that ∑_i w_i=1.
10. Adjust w such that max_i w_i<=0.2 to avoid skewing.
11. for i=1..n
12.   Normalize t_i such that ||t_i||=w_i.
13. endfor
```

5 Experiments

5.1 Experimental Setup

We conducted the proposed approach on three infrared (IR) sequences (V3V300004_003, V3V300004_004 and V3V300013_012) selected from the VIVID database [21]. All sequences contain several vehicles driving on road. The videos are taken from air and vehicles in the videos change their pose (turning on the road) between frames.

In addition to the proposed method, we tested the original L_1 tracker [17] and the *mean-shift tracker* [5] on the sequences. All three methods share the same manual initialization. For the mean shift tracker, which does not deal with pose change, the manual initialization is squared

(but has a same area) to add robustness. For both L_1 regularization methods, we use same parameter settings whenever possible. For example, for both trackers we use n=10 (i.e., 10 templates), =0.01 (for L_1 regularization), 100 particles, and same template sizes. Other parameters used in eL_1-PF include $_r$ = 0.1, = 0.2.

5.2 Experimental Result

Figures 3, 4 and 5 show the tracking results on the three IR sequences respectively. In each figure, the tracked targets are displayed with red bounding boxes (i.e., in the state space). Six frames across vehicle turning are selected to illustrate the performances of different trackers.

From the results, we can see clearly that the new approach achieves the best performance and is robust to pose change. Some specific observations are listed below:

- Both mean shift (MS) and the original L_1 (L_1-PF) trackers fail on the first sequence (V3V300004_003) as shown in Figure 3. For MS, the failure starts when the target is close to surroundings that share a similar intensity distribution with the target. This is observed in the second and third rows. The original L_1 tracker survives at this point, but meets problems when the vehicle starts to turn around. Rows three and four show that, the shape of the original L_1 tracker is tuned to the wrong direction, which is mainly due to that its affine state space enforces little shape constraints. In contrast, the proposed new eL_1–PF tracker successfully tracks target and at the same time accurately estimates its pose.
- All three methods perform reasonably well on the second sequence (V3V300004_004) as shown in Figure 4. This is mainly because the target vehicle has a square shape. In this case, the pose change does not raise big issues to the mean shift tracker, which relies mainly on the intensity distributions. Through a detailed check, however, we can still see that the proposed eL_1–PF tracker method generates a better pose estimation.
- On the third sequence (V3V300013_012) presented in Figure 5, we see that the both L_1 trackers again outperform the mean shift tracker. The original L_1 tracker failed to correctly estimate the target pose, which is correctly captured by our eL_1–PF tracker approach.

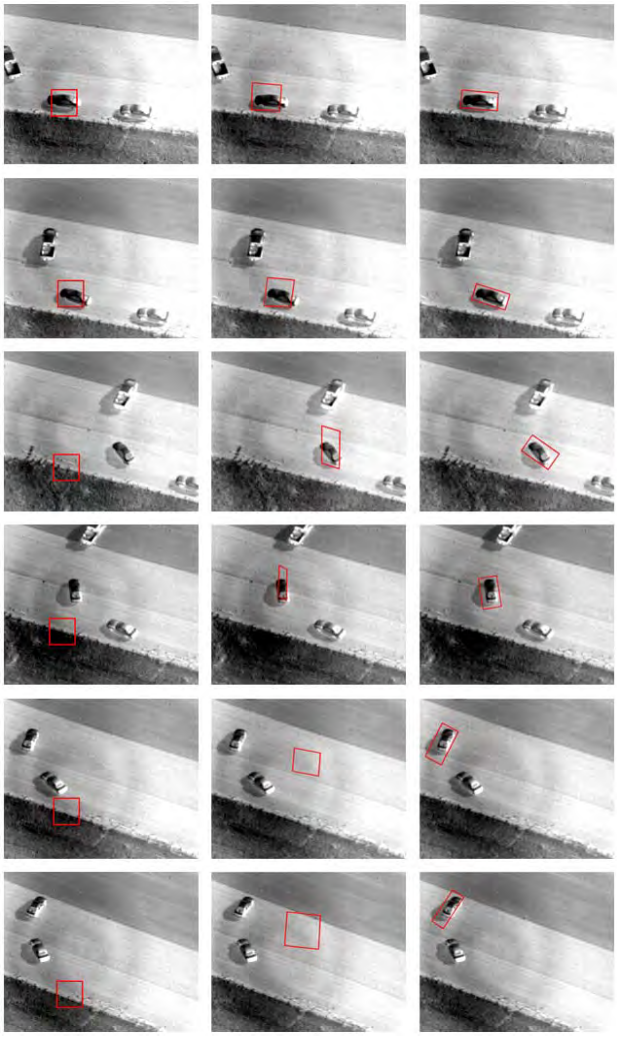
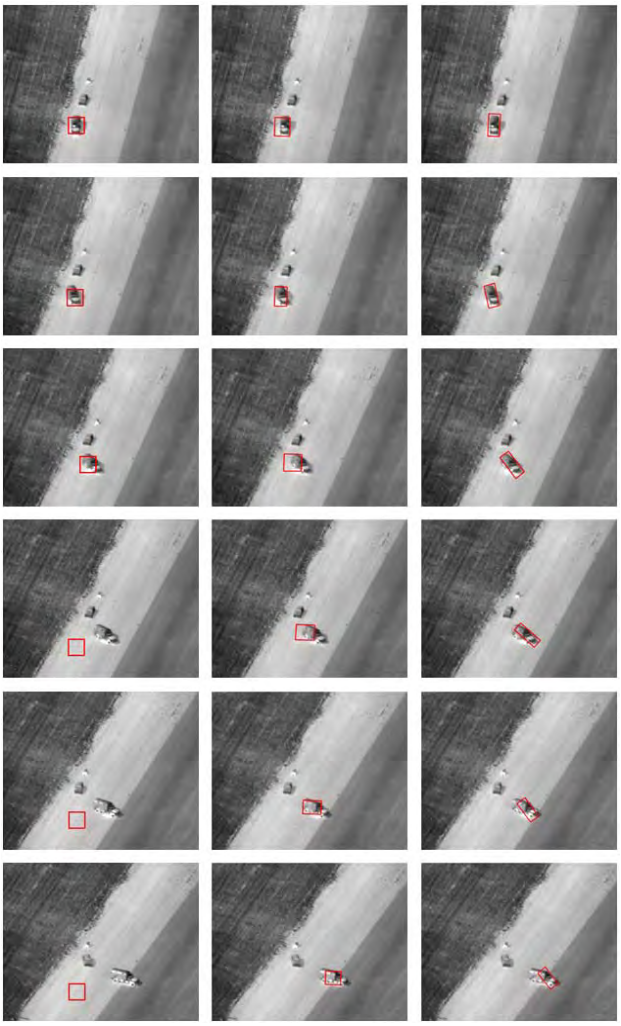
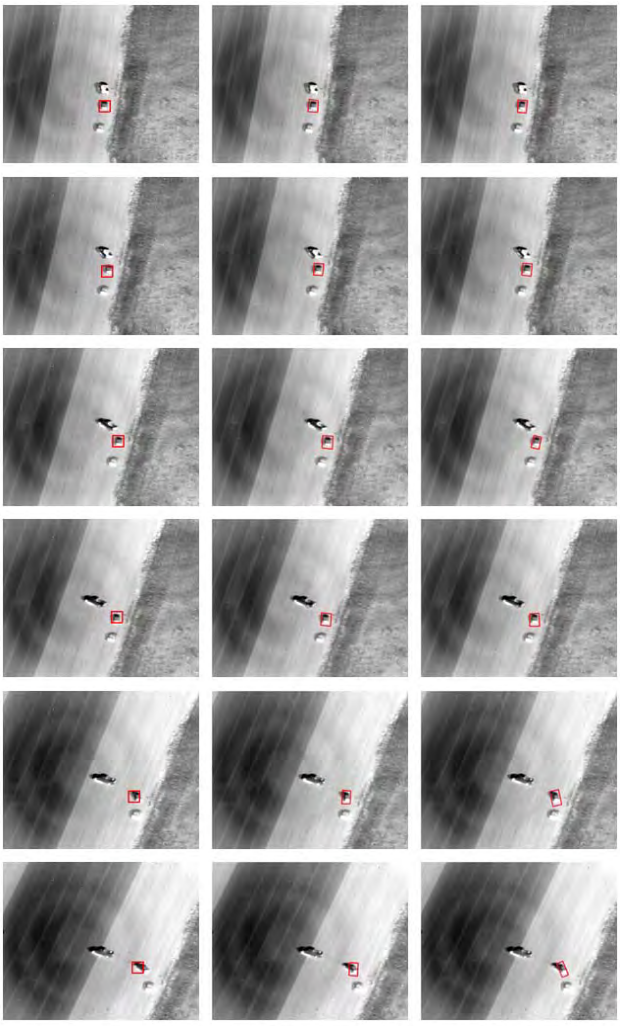


Figure 3. Tracking results on VIVID IR sequence V3V300004_003. Frame from top to bottom: 1451, 1483, 1515, 1547, 1579 and 1611. Trackers, from left to right, are the mean shift tracker [5], the original L_1 tracker [17], and the proposed new eL_1–PF tracker.



Meanshift **L_1–PF** **eL_1–PF**

Figure 5. Tracking results on VIVID IR sequence V3V300013_012. Frame from top to bottom: 0020, 0052, 0084, 0116, 0148 and 0180. Trackers are same as in the previous figure.



Meanshift **L_1–PF** **eL_1–PF**

Figure 4. Tracking results on VIVID IR sequence V3V300004_004. Frame from top to bottom: 1001, 1033, 1065, 1097, 1129 and 1161. Trackers are same as in the previous figure.

6 Conclusion

We present a new visual tracker by exploiting the sparseness intrinsic in the tracking process. As a result, the new method uses L_1 regularization in the particle filter framework, armed with a modeling for target pose and a probabilistic template updating. The *extended L_1-regularized particle-filter* (eL_1-PF) method is applied to vehicle tracking in IR videos involving serious pose changes. Experiments on the VIVID dataset show the superiority of the proposed method over the previously proposed mean shift and L_1–PF approaches.

In the future, we plan to focus on several directions along the work. First, the current L_1 tracker does not take benefit of the dependences between particle samples, which we believe can be used for improving the tracking efficiency. Second, the method can be naturally extended to multi-target tracking and multi-modality tracking, since the representation has little limitation on the input sequences. Lastly, we can incorporate the multi-modality into simultaneous tracking and identification scenarios to discriminate between like targets, which has many important applications [2][18][19] [27].

Acknowledgement

We thank the reviewers for insightful suggestions. Haibin Ling is supported in part by NSF (Grant No. IIS-0916624). Xue Mei contributed to this work when he was with the University of Maryland College Park. This

material is based upon research work partially supported by the Office of Naval Research (ONR) Grant N00014-09-C-0070. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the ONR.

7 References

[1] M. J. Black and A. D. Jepson. "Eigentracking: Robust matching and tracking of articulated objects using a view-based representation," *International Journal of Computer Vision*, 26:63-84, 1998.

[2] E. Blasch and B. Kahler, "Multiresolution EO/IR Target Track and Identification", *ISIF Proc. Fusion05*, 2005.

[3] Y. Boykov and D. Huttenlocher. "Adaptive bayesian recognition in tracking rigid objects," in *Proc. of the Computer Vision and Pattern Recognition* (CVPR), 697--704, 2000.

[4] E. Candës, J. Romberg, and T. Tao. "Stable signal recovery from incomplete and inaccurate measurements," *Comm. on Pure and Applied Math*, 59(8):1207-1223, 2006.

[5] D. Comaniciu, V. Ramesh, and P. Meer. "Kernel-based object tracking," *IEEE Trans on Pattern Anal. and Mach. Intell.*, 25:564-577, 2003.

[6] D. Donoho. "Compressed Sensing," *IEEE Trans. Information Theory*, 52(4):1289-1306, 2006.

[7] A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 2001, New York.

[8] G.J. Edwards, C.J. Taylor, and T.F. Cootes. "Improving Identification Performance by Integrating Evidence from Sequences," *Proc. of the Computer Vision and Pattern Recognition* (CVPR), 1:486-491, 1999.

[9] G. D. Hager and P. N. Belhumeur. "Efficient region tracking with parametric models of geometry and illumination," *IEEE Trans on Pattern Anal. and Mach. Intell.*, 20:1025-1039, 1998.

[10] J. Ho, K.-C. Lee, M.-H. Yang, and D. Kriegman. "Visual tracking using learned subspaces," *Proceedings of the Computer Vision and Pattern Recognition* (CVPR), 782-789, 2004.

[11] M. Isard and A. Blake. "Condensation - conditional density propagation for visual tracking," *International Journal of Computer Vision*, 29:5-28, 1998.

[12] Z. Khan, T. Balch, and F. Dellaert. "A Rao-Blackwellized particle filter for EigenTracking," *Proceedings of the Computer Vision and Pattern Recognition* (CVPR), 980-986, 2004.

[13] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky. "A method for large-scale l_1-regularized least squares," *IEEE Journal on Selected Topics in Signal Processing*, 1(4):606-617, 2007.

[14] B. Lucas and T. Kanade. "An iterative image registration technique with an application to stereo vision," *International Joint Conferences on Artificial Intelligence* (IJCAI), 674-679, 1981.

[15] I. Matthews, T. Ishikawa, and S. Baker. The template update problem, *IEEE Trans on Pattern Anal. and Mach. Intell.*, 810-815, 2004.

[16] X. Mei, H. Ling, and D.W. Jacobs. "Sparse Representation of Cast Shadows via l_1-Regularized Least Squares," *Proceedings of the International Conference on Computer Vision* (ICCV), 2009.

[17] X. Mei and H. Ling. "Robust Visual Tracking using l_1 Minimization," *Proceedings of the International Conference on Computer Vision* (ICCV), 2009.

[18] P. Minviell, A. D. Marrs, S. Maskell, and A. Doucet, "Joint Target Tracking and Identification – Part I: Sequential Monte Carlo Model-Based Approaches, *ISIF Proc. Fusion05*, 2005.

[19] P. Minviell, A. D. Marrs, S. Maskell, and A. Doucet, "Joint Target Tracking and Identification – Part II: Shape Video Computing, *ISIF Proc. Fusion05*, 2005.

[20] D. A. Ross, J. Lim, R. Lin and M. Yang. "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, 77:125-141, 2008.

[21] VIVID database. [online] https://www.sdms.afrl.af.mil/request/data_request.php#vivid

[22] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. "Robust Face Recognition via Sparse Representation," *IEEE Trans on Pattern Anal. and Mach. Intell.*, 31(1):210-227, 2009.

[23] C. Yang and E. Blasch, "Pose Angular-Aiding for Maneuvering Target Tracking," *ISIF Proc. Fusion05*, 2005.

[24] C. Yang and E. Blasch. "Kalman Filtering with Nonlinear State Constraints." *ISIF Proc. Fusion 07, 2007*.

[25] A. Yilmaz, O. Javed, and M. Shah. "Object tracking: A survey." *ACM Comput. Survey*, 38(4), 2006.

[26] S. K. Zhou, R. Chellappa, and B. Moghaddam. "Visual tracking and recognition using appearance-adaptive models in particle filters," *IEEE Trans. Image Processing*, 11:1491-1506, 2004.

[27] E. Blasch and L. Hong "Simultaneous Identification and Track Fusion," *IEEE Conf. on Dec. Control*, Orlando, FL, Dec 1998.